



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/873,730	06/02/2001	Gerald Neufeld	4906.P021	4760

7590 04/25/2005

Daniel M. DeVos
Blakely, Sokoloff, Taylor & Zafman LLP
Seventh Floor
12400 Wilshire Boulevard
Los Angeles, CA 90025-1030

EXAMINER

PHAM, HUNG Q

ART UNIT	PAPER NUMBER
----------	--------------

2162

DATE MAILED: 04/25/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/873,730

Applicant(s)

NEUFELD ET AL.

Examiner

HUNG Q PHAM

Art Unit

2162

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 31 January 2005.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-20, 22-50, 52 and 53 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-20, 22-50, 52 and 53 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152) |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

A

DETAILED ACTION

Response to Arguments

- Applicant's arguments filed 01/31/2005 with respect to the rejection of claims 2, 7, 8, 24, 30, 38, 43 and 44 under 35 U.S.C. § 112, first paragraph, at pages 10-12 have been fully considered but they are not persuasive.

(1) As in the REMARKS filed on 06/28/2004, Applicants referred to paragraph [0047] as the written description of the claimed *executing the operation idempotently with a network resource process*.

As illustrated in paragraph [0047], an idempotent operation as claimed is an operation that can be executed several times with the same results as if it succeeds on the first execution. Applicants also pointed out that the CLI operation is executed idempotently with a network resource process.

In order to prove claims 2 and 38, *moving the record into a database atomically*, are supported by the Specification, Applicants refer examiner to paragraph [0027] for the description of claims 2 and 38 as in the REMARKS filed on 01/31/2005:

Applicants believe that the Examiner misunderstands the disclosure and Applicants' arguments. Applicants refer the Examiner to paragraph [0027] to show that the movement of records into the database after receiving a commit command may be done atomically with an atomic insert operation.

Assuming the examiner misunderstood the disclosure, however, examiner respectfully points out that the illustration of [0027] is the process of inserting a recorded RDB operation, not for the claimed operation that is *executed idempotently with a network resource process*. As illustrated at FIG. 1, and in the disclosure below:

[0021] The RCM 101 receives transactions and stores operations from these transactions in the log 110 via the RDB 109... The RCM 101 receives transactions from a set of one or more command line interfaces (CLIs) 103... Operations from a CLI 103 will be referred to as a CLI operation... A component manager 111 processes a CLI operation into one or more RDB operations. An RDB operation is either a backend procedure call or a database record request. An example of a CLI operation would be the command "start OSPF" entered by a user at a CLI... Each RDB operation is stored within the stable log memory 110 until a commit command for the transaction is received. Once the commit command for the transaction is received, the corresponding RDB operation(s) are performed (the backend procedure call and the database record update request)...

[0026] In another embodiment, the RDB is implemented as a binary search tree...

[0027] For every request to add or delete a node from the binary tree, all nodes surrounding the object to be changed are replicated...

Thus, the limitation of claims 2 and 38 is for RDB operations, and RDB operation is totally different from CLI because RDB operation is not an idempotent operation. In other words, RDB is not *executed idempotently with a network resource process*, because *once the commit command for the transaction is received, the corresponding RDB operation(s) are performed (the backend procedure call and the database record update request)*. In short, no record that stores CLI commands to be moved into a database atomically.

(2) As argued by Applicants at page 10:

The Examiner has already noted that paragraph [0021] sets forth that the RCM receives transactions and stores operations from these transactions in a log. The Examiner also notes that an example of an operation that might be stored in these logs could be a CLI operation such as the "start OSPF" operation. The Examiner failed to note that paragraph [0021] also goes on to state that "each RDB operation is stored within the stable log memory 110 until a commit command for the transaction is received."

....

Thus, the "start OSPF" command cited as an example in paragraph [0021] and noted by the Examiner might be an operation that is stored as a record in the log of the RDB. When that "start OSPF" command is saved in the log, the operation of saving the command is atomic. Subsequently, a commit command may be received and the "start OSPF" command in the log may be processed. Thus, the elements of claim 2 are supported by the specification.

Examiner respectfully traverses because of the following reasons:

Assuming examiner failed to note that paragraph [0021] also goes on to state that "each RDB operation is stored within the stable log memory 110 until a commit command for the transaction is received", but examiner cannot note that an example of an operation that might be stored in these logs could be a CLI operation such as the "start OSPF" operation, and the "start OSPF" command cited as an example in paragraph [0021] and noted by the Examiner might be an operation that is stored as a record in the log of the RDB. When that "start OSPF" command is saved in the log, the operation of saving the command is atomic if there is no written description in such full, clear, concise, and exact terms as set forth in 35 U.S.C. 112, first paragraph, and examiner cannot assume "start OSPF" command is saved in the log, the operation of saving the command is atomic, and [S]ubsequently, a commit command may be received and the "start OSPF" command in the log may be processed when there is no written description in such full, clear, concise, and exact terms.

(3) Applicant's arguments with respect to the rejection of claims 7 and 43 under 35 U.S.C § 112, first paragraph, have been fully considered and are persuasive.

Art Unit: 2162

The rejection of claims 7 and 43 under 35 U.S.C § 112, first paragraph, has been withdrawn.

(4) As argued by Applicants at pages 11-12 with respect to the rejection of claims 8 under 35 U.S.C § 112, first paragraph, examiner respectfully traverses because the referred paragraph [0032] and FIG. 4A, 4B and 5 is the process of determining lock contention. Although CLI operations can be grouped, but there is no description of performing idempotently each of the sequence of CLI operations, e.g., as in FIG. 5, if CLI is an abort command, this CLI operation is aborted and not executed several times with the same results as if it succeeds on the first execution, and there is no network resource process associated with CLI abort.

- Applicants arguments with respect to the rejection of claims 1-2, 4-5, 7-8, 10, 12-14, 16, 24, 26, 29-30, 32-34, 36-38, 40-41, 43-44 and 46 under 35 U.S.C § 102 have been fully considered but they are not persuasive.

(1) As argued by applicants at page 12:

In regard to independent claims 1, 24 and 37, these claims includes the elements of "storing an operation" and "executing the operation idempotently with a network resource process." The Examiner cites col. 19, line 9-15 as teaching these elements of claim 1. Specifically, the Examiner relies on the "end commit ()" method which is idempotent for teaching these elements of claims 1, 24 and 37. However, the "end commit" method is not an operation that is stored in the log of Long and then executed idempotently. Rather, it is a commit process which operates on the log. See generally, col. 18, line 52 through col. 19, line 15 which discusses a method that operates on the log received from the CRM worker. Thus, the

Examiner has not established that Long teaches "storing an operation" and "executing an operation idempotently with a network resource process" as claimed in claims 1, 24 and 37. Accordingly, reconsideration and withdrawal of the anticipation rejection of claims 1, 24 and 37 are requested.

Claims 2, 4, 5, 26, 38, 40 and 41 depend from independent claims 1, 24 and 37 and incorporate the limitations thereof. Thus, at least for the reasons mentioned above in regard to independent claims 1, 24 and 37, these claims are not anticipated by Long. Accordingly, reconsideration and withdrawal of the anticipation rejection of claims 2, 4, 5, 26, 38, 40 and 41 are requested.

Examiner respectfully traverses because of the following reasons:

As disclosed by Long, the clean-up action implemented in the EndCommit () method should be idempotent, so that it may be attempted multiple times with the same results as if it succeeds on the first attempt. Such as in a case where a failure occurs after the CRM compensator performs the clean-up action but before logging that the clean-up action was done (Col. 19, Lines 9-15). The CRM compensator also uses the lcrmLogControl interface to log the clean-up actions for recovery purposes (Col. 18, Line 66-Col. 19, Line 1). Log file and log records are illustrated at Col. 13, Lines 57-65). As seen, clean-up action as *the operation is executed idempotently with* EndCommit () as *a network resource process*, and logging the clean-up action for recovery purpose implies the step of *storing clean-up action as an operation in a log as a record*.

Claims 2, 4, 5, 26, 38, 40 and 41 depend from independent claims 1, 24 and 37, and theses claims are anticipated by Long as sets forth in the Action.

(2) As argued by applicants at page 13:

Long does not teach managing data for transactions in a network element. Rather, Long teaches a system of transaction processing for a server application and client environment. See col. 5, lines 40-52. Thus, Long does not teach each of the elements of claims 7, 12 and 43. Accordingly, reconsideration and withdrawal of the anticipation rejection of claims 7, 12 and 43 are requested.

In regard to claims 8, 10, 13, 14, 16 and 46, these claims depend from independent claims 7, 12 and 43 and incorporate the limitations thereof. Thus, at least for the reasons mentioned above in regard to independent claims 7, 12 and 43, these claims are not anticipated by Lone. Accordingly, reconsideration and withdrawal of the anticipation rejection of these claims are requested.

Examiner respectfully traverses because of the following reasons:

As illustrated at FIG. 2, a server-client network is disclosed (Col. 9, Lines 48-62).

As illustrated in FIG. 4, when server application component 86 has a transaction, the CRM worker logs sufficient information at step 255 to the persistent log using the CRM clerk to be able to take appropriate clean-up and compensating action (Col. 14, Line 45-Col. 15, Line 34). The log file and log records is disclosed at Col. 13, Lines 47-65 and stored in the server 20. Two phases commit protocol notifications are issued in response to SetComplete indicating that all works in the transaction is completed, and SetAbort indicating the works cannot be completed for performing clean-up action on commit, or compensating action on abort, wherein clean-up and compensating actions are logged by CRM compensator (Col. 15, Line 35-Col. 16, Line 3). Long further discloses that a transaction is a collection of actions that conform to a set of properties, which include atomicity. Atomicity means that all activities in a transaction either take effect together as a unit, or all fail (Col. 1, Lines 43-47). As seen, the logging technique,

Art Unit: 2162

and the sequence action such as *normal action* → *clean-up action*, or *normal action* → *compensating action* indicates the step of *storing a sequence of operations* at server 20 as *a network element*, and the atomicity of a transaction include actions as discussed above indicates the step of *performing the sequence of operation as an atomic transaction*.

Claims 8, 10, 13, 14, 16 and 46 are dependent claims, and these claims also are rejected as sets forth in the Action.

(3) As argued by Applicant at page 13:

Claims 29 and 33 relates to a network element containing a processor and a storage unit. The Examiner states on page 12 that "Long teaches a network element as in Figure 1." However, Figure 1 does not depict a network element. Rather, Figure 1 of Long clearly teaches a server computer 20. See Figure 1 of Long. Thus, Long does not teach a process or a storage unit in a network element as claimed in claims 29 and 33. Accordingly, reconsideration and withdrawal of the anticipation rejection of claims 29 and 33 are requested.

In regard to claims 30, 32, 34 and 36, these claims depend from independent claims 29 and 33 incorporate the limitations thereof. Thus, at least for the reasons mentioned above in regard to independent claim 29 and 33, these claims are not anticipated by Long. Accordingly, reconsideration and withdrawal of the anticipation rejection of claims 30, 32, 34 and 36 are requested.

Examiner respectfully traverses because as in FIG. 1, server computer 20 is *a network element* of sever-client network as illustrated in FIG. 2, Col. 9, Lines 48-62. Server 20 of FIG. 1 has *a processor* (PROCESSING UNIT 21) and HARD DRIVE 27 as *a storage unit coupled to the processor*.

Art Unit: 2162

Claims 30, 32, 34 and 36 are dependent claims, and these claims also are rejected as sets forth in the Action.

- Applicants arguments with respect to the rejection of claims 3, 9, 15, 18-20, 23, 25, 31, 35, 39, 45, 58-50 and 53 under 35 U.S.C § 103 have been fully considered but they are not persuasive because of the reasons as discussed above with respect to the teaching of Long, and the technique of lock contention as taught by Traversat is a must for Long system in order to control concurrency access.

Claim Rejections - 35 USC § 112

The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

Claims 2, 7-8, 24, 30, 38 and 43-44 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter, which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

As in claims 2 and 38, the steps of *moving the record into a database atomically*, and *each of the sequence of operations is performed idempotently* as in claims 8 and 44, *storing a first and second operation as an atomic transaction; performing the first and second operation idempotently with a set of network resource process* as in claim 24, *executing each of the set of operations idempotently* as in claim 30 were not described in the specification.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371(c) of this title before the invention thereof by the applicant for patent.

The changes made to 35 U.S.C. 102(e) by the American Inventors Protection Act of 1999 (AIPA) and the Intellectual Property and High Technology Technical Amendments Act of 2002 do not apply when the reference is a U.S. patent resulting directly or indirectly from an international application filed before November 29, 2000. Therefore, the prior art date of the reference is determined under 35 U.S.C. 102(e) prior to the amendment by the AIPA (pre-AIPA 35 U.S.C. 102(e)).

Claims 1, 2, 4, 5, 7, 8, 10, 12-14, 16, 24, 26, 29, 30, 32-34, 36-38, 40, 41, 43, 44 and 46 are rejected under 35 U.S.C. 102(e) as being anticipated by Long [USP 6,526,416 B1].

Regarding claims 1 and 37, Long teaches a machine readable medium that provides instructions, which when executed by a set of processors of one or more processors, cause said set of processors to perform operations (FIGS. 1 and 2). As illustrated at box 251 of FIG. 4, the server application component 86 has a transaction (Col. 14, Lines 45-55). At box 254, the server application component 86 requests for work to be performed on the resource as part of its transaction (Col. 15, Lines 21-23), and two phase-commit protocol notifications to participants in the transaction are issued at box 256 (Col. 15, Lines 35-41). As further disclosed by Long, the clean-up action implemented in the EndCommit () method should be idempotent, so that it may be attempted multiple times with the same results as if it succeeds on the first attempt. Such as in a case where a failure occurs after the CRM compensator performs the clean-up action but before logging that the clean-up action was done (Col. 19, Lines 9-15). The CRM compensator also uses the lcrmLogControl interface to log the clean-up actions for recovery purposes (Col. 18, Line 66-Col. 19, Line 1). Log file and log records are illustrated at Col. 13, Lines 57-65). As seen, clean-up action as *the operation is executed idempotently with* EndCommit () as *a network resource process*, and logging the clean-up action for recovery purpose implies the step of *storing clean-up action as an operation in a log as a record*.

Regarding claims 2 and 38, Long teaches all the claim subject matters as discussed above with respect to claims 1 and 37, Long further discloses the step of

Art Unit: 2162

storing the operation in a log as a record; receiving a commit command (Col. 18, Line 52-Col. 19, Line 15), *moving the record into database atomically* (Col. 13, Line 57-Col. 14, Lines 4).

Regarding claims 4 and 40, Long teaches all of the claimed subject matter as discussed above with respect to claims 1 and 37, Long further discloses *the operation is one of a sequence of operations comprising an atomic transaction* (Col. 1, Lines 43-61).

Regarding claims 5 and 41, Long teaches all of the claimed subject matter as discussed above with respect to claims 1 and 37, Long further discloses the steps of *receiving the operation from a first user; and receiving a second operation from a second user* (Col. 1, Lines 14-19).

Regarding claims 7 and 43, Long teaches a machine-readable medium that provides instructions, which when executed by a set of processors of one or more processors, cause said set of processors to perform operations (FIGS. 1). As illustrated at FIG. 2, a server-client network is disclosed (Col. 9, Lines 48-62). As illustrated in FIG. 4, when server application component 86 has a transaction, the CRM worker logs sufficient information at step 255 to the persistent log using the CRM clerk to be able to take appropriate clean-up and compensating action (Col. 14, Line 45-Col. 15, Line 34). The log file and log records is disclosed at Col. 13, Lines 47-65 and stored in the server 20. Two phases commit protocol notifications are issued in response to SetComplete indicating that all works in the transaction is completed, and SetAbort indicating the

Art Unit: 2162

works cannot be completed for performing clean-up action on commit, or compensating action on abort, wherein clean-up and compensating actions are logged by CRM compensator (Col. 15, Line 35-Col. 16, Line 3). Long further discloses that a transaction is a collection of actions that conform to a set of properties, which include atomicity. Atomicity means that all activities in a transaction either take effect together as a unit, or all fail (Col. 1, Lines 43-47). As seen, the logging technique, and the sequence action such as *normal action* → *clean-up action*, or *normal action* → *compensating action* indicates the step of *storing a sequence of operations* at server 20 as *a network element*, and the atomicity of a transaction include actions as discussed above indicates the step of *performing the sequence of operation as an atomic transaction*.

Regarding claims 8 and 44, Long teaches all of the claimed subject matter as discussed above with respect to claims 7 and 43, Long further discloses *each of the sequence of operations is performed idempotently* (Col. 4, Lines 30-37).

Regarding claims 10 and 46, Long teaches all of the claimed subject matter as discussed above with respect to claims 7 and 43, Long further discloses *a sequence of operations is received from a first user and a second sequence of operations is received from a second user* (Col. 1, Lines 14-19).

Regarding claim 12, Long teaches a machine readable medium that provides instructions, which when executed by a set of processors of one or more processors,

Art Unit: 2162

cause said set of processors to perform operations (FIGS. 1 and 2). As illustrated at FIG. 4, the CRM worker logs sufficient information at step 255 to the persistent log using the CRM clerk to be able to take appropriate clean-up and compensating action (Col. 14, Line 45-Col. 15, Line 34). The clean-up action as *operation* is logged in a log file (Col. 18, Line 66-Col. 19, Line 1). Each log record is identified by a log sequence number (Col. 13, Lines 57-64). Log file is stored in hard drive 27 of server 20 (FIG. 1, Col. 13, Lines 57-60). In other words, the technique of logging clean-up action using log sequence number indicates the step of *storing atomically an operation in database of server 20 as network element atomically*. In response to SetComplete as *a commit command*, the clean-up action is implemented in the EndCommit () method as *network resource process* (Col. 19, Lines 9-15). In different words, the Long technique as discussed indicates the claimed *performing the operation with a network resource process in response to a commit command*.

Regarding claim 13, Long teaches all of the claimed subject matter as discussed above with respect to claim 12, Long further discloses *the operation is performed idempotently* (Col. 19, Lines 9-15).

Regarding claim 14, Long teaches all of the claimed subject matter as discussed above with respect to claim 12, Long further discloses *the operation is one of a sequence of operations comprising a transaction* (Col. 14, Line 45-Col. 1, Line 3).

Regarding claim 16, Long teaches all of the claimed subject matter as discussed above with respect to claim 12, Long further discloses *operations is received from a first user and a second operations is received from a second user* (Col. 1, Lines 14-19).

Regarding claim 24, Long teaches a machine-readable medium that provides instructions, which when executed by a set of processors of one or more processors, cause said set of processors to perform operations (FIGS. 1 and 2). As illustrated in FIG. 4, when server application component 86 has a transaction, CRM worker object is created and associated with the transaction. CRM worker creates an instance of CRM clerk at box 252, and CRM worker perform work on the resource as part of its transaction. As the CRM worker performs the request work, i.e., normal action, the CRM worker logs sufficient information at step 255 to the persistent log using the CRM clerk to be able to take appropriate clean-up and compensating action (Col. 14, Line 45-Col. 15, Line 34). Two phases commit protocol notifications are issued in response to SetComplete indicating that all works in the transaction is completed, and SetAbort indicating the works cannot be completed for performing clean-up action on commit, and compensating action on abort, wherein clean-up and compensating actions are logged by CRM compensator (Col. 15, Line 35-Col. 16, Line 3). As seen, the sequence action such as *normal action → clean-up action*, or *normal action → compensating action* indicates *a first operation*, and *a second operation* are stored in a log file for implementing two phases commit protocol. And the purpose of two phases commit protocol is to maintain the atomicity of the transaction. In different words, the

Art Unit: 2162

technique as discussed indicates the step of *storing a first and second operation as an atomic transaction*. Long further discloses the clean-up action is implemented in the EndCommit () method as *network resource process* should be *idempotent*, and the idempotent compensating action is implemented in EndAbort () method as *network resource process* (Col. 19, Lines 9-15). In different words, the Long technique as discussed indicates the claimed *performing the first and second operation idempotently with a set of network resource processes*.

Regarding claim 26, Long teaches all the claim subject matters as discussed above with respect to claim 24, Long further discloses the steps of *storing the first and second operation of the atomic transaction in a log* (Col. 14, Line 45-Col. 15, Line 34 and Col. 15, Line 35-Col. 16, Line 3), *receiving a commit command for the atomic transaction; indicating the atomic transaction as committed* (Col. 18, Line 52-Col. 19, Line 34), and *storing the atomic transaction in a database* (Col. 16, Lines 39-55 and Col. 19, Line 28-35).

Regarding claim 29, Long teaches a network element as in FIG. 1 with *a processor and a storage unit coupled to the processor* (Col. 5, Line 55-Col. 7, Line 56). The processor is *to execute a set of atomic transactions* as illustrated at FIG. 4 (Col. 1, Lines 43-47 and Col. 14, Line 45-Col. 16, Line 3, and the storage unit is *to store the set of atomic transaction* as illustrated at FIG. 5 (Col. 16, Lines 5-55).

Regarding claim 30, Long teaches all the claim subject matters as discussed above with respect to claim 29, Long further discloses *the set of atomic transactions is comprised of a set of operations* (Col. 15, Line 21-Col. 16, Line 3), *the processors to execute each of the set of operations idempotently* (Col. 18, Line 28-Col. 19, Line 35).

Regarding claim 32, Long teaches all the claim subject matters as discussed above with respect to claim 29, Long further discloses *a set of interfaces coupled to the processor, each of the set of interfaces to receive at least one of the set of atomic transactions, each of the set of interfaces corresponding to a different user* (Col. 6, Line 62-Col. 7, Line 56).

Regarding claim 33, Long teaches a network element as illustrated at FIG. 1 comprises *an interface to receive a plurality of operations from a user* (Col. 6, Line 62-Col. 7, Line 55), *a configuration manager coupled to the interface, the configuration manager to process the plurality of operations* (Transaction Manager, Col. 10, Lines 28-45), and *a database coupled to the configuration manager, the database to store the plurality of operations* (Col. 13, Line 57-Col. 14, Line 4) *as an atomic transaction* (Col. 10, Lines 33-43).

Regarding claim 34, Long teaches all the claim subject matters as discussed above with respect to claim 33, and further discloses the step of *processing the plurality of operations as atomic transactions* (Transaction Manager, Col. 10, Lines 28-45).

Art Unit: 2162

Regarding claim 36, Long teaches all the claim subject matters as discussed above with respect to claim 33, and further discloses *a second interface to receive a second plurality of operations from a second user* (Col. 6, Line 62-Col. 7, Line 21).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

Claims 3, 9, 15, 18-20, 23, 25, 31, 35, 39, 45, 48-50 and 53 are rejected under 35 U.S.C. 103(a) as being unpatentable over Long [USP 6,526,416 B1] in view of Traversat et al. [USP 6,115,715].

Art Unit: 2162

Regarding claims 3 and 39, Long teaches all the claimed subject matters as discussed in claims 1 and 37, Long further discloses the step of *receiving the operation* (Col. 18, Line 52-Col. 19, Line 15), but fails to disclose the steps of *performing lock contention handling for the operation; storing the operation if a lock contention is not detected; and generating a lock contention notification if the lock contention is detected for the operation.*

Traversat teaches a method for updating and managing a configuration database, and further discloses the steps of *performing lock contention handling for the operation; storing the operation if a lock contention is not detected; and generating a lock contention notification if the lock contention is detected for the operation* (Col. 8, Line 3-Col. 9, Line 65). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to include the step of performing lock handling in order to execute the operation in a stable and correct state.

Regarding claims 9 and 45, Long teaches all of the claimed subject matter as discussed above with respect to claims 7 and 43, but does not disclose the steps of *performing lock contention handling for each of the sequence of the operation; storing the sequence of the operation if a lock contention is not detected; generating a lock contention notification if the lock contention is detected.* Traversat teaches a method for updating and managing a configuration database, and further discloses the steps of *performing lock contention handling for each of the sequence of the operation; storing the sequence of the operation if a lock contention is not detected; generating a lock contention notification if the lock contention is detected* (Col. 8, Line 3-Col. 9, Line 65). Therefore, it would have been obvious for one of

Art Unit: 2162

ordinary skill in the art at the time the invention was made to include the step of performing lock handling in order to execute the operation in a stable and correct state.

Regarding claim 15, Long teaches all of the claimed subject matter as discussed above with respect to claim 12, but does not disclose the steps of *performing lock contention handling for the operation; storing the operation in the database if a lock contention is not detected; and generating a notification of lock contention if the lock contention is detected*. Traversat teaches a method for updating and managing a configuration database, and further discloses the steps of *performing lock contention handling for the operation; storing the operation in the atomic database if a lock contention is not detected; and generating a notification of lock contention if the lock contention is detected* (Col. 8, Line 3-Col. 9, Line 65). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to include the step of performing lock handling in order to execute the operation in a stable and correct state.

Regarding claims 18 and 48, Long teaches a machine-readable medium that provides instructions, which when executed by a set of processors of one or more processors, cause said set of processors to perform operations (Long, FIGS. 1 and 2). As illustrated at box 251 of FIG. 4, when server application component 86 has a transaction, CRM worker object is created and associated with the transaction. CRM worker creates an instance of CRM clerk at box 252, and CRM worker perform work, normal action, on the resource as part of its transaction (Long, Col. 14, Line 45-Col. 15,

Art Unit: 2162

Line 34). As seen, the normal action as *an operation is received* at the resource. Two phases commit protocol notifications are issued in response to SetComplete indicating that all works in the transaction is completed, or SetAbort indicating the works cannot be completed for performing clean-up action on commit, or compensating action on abort (Long, Col. 15, Line 35-Col. 16, Line 3). Long further discloses that a transaction is a collection of actions that conform to a set of properties, which include atomicity.

Atomicity means that all activities in a transaction either take effect together as a unit, or all fail (Long, Col. 1, Lines 43-47). As seen, *the sequence action*, such as *normal action* → *clean-up action*, or *normal action* → *compensating action*, contains normal action as *one operation of the sequence*, and normal action is *an atomic transaction* because normal action is completed if all works of normal action is completed, and normal action will not be completed if the works cannot be completed. In other words, the technique as discussed performs the claimed *the operation is one of a sequence of operation comprising an atomic transaction*. Long does not teach the step of *determining if a lock contention exists for a record corresponding to the operation*, and *generating a notification of the lock contention if a lock contention does exist for the record*. Traversat teaches a method for updating and managing a configuration database. As shown in FIG. 5, at step 502, the transaction of adding a new printer to a client will attempt to lock the appropriate node in a sub-tree of the client JSD schema (Traversat, Col. 8, Lines 3-5). If the attempt to lock the appropriate parent node or leaf node fails, control goes to step 504 (Traversat, Col. 8, Lines 5-7) as the step of *determining if a lock contention exists for a record corresponding to the operation*. At step 504 the system determines whether the transaction is a blocking or unblocking

Art Unit: 2162

transaction (Traversat, Col. 8, Lines 7-8). If the transaction is blocking, control returns to step 502 where the transaction attempts to acquire a lock on the desired entry, this time after waiting in a queue and being notified by an event manager (Traversat, Col. 8, Lines 35-38). As seen, the transaction is stored in a queue and being notified when a lock exist for the node. In other words, this performs the step of *generating a notification of the lock contention if a lock contention does exist for the record*. It would have been obvious for one of ordinary skill in the art at the time the invention was made to include the technique of locking and generating lock notification as taught by Traversat in order to manage database concurrent access.

Regarding claims 19 and 49, Long and Traversat, in combination, teach all of the claimed subject matter as discussed above with respect to claims 18 and 48, Long further discloses *the operation is one of a sequence of operations comprising an atomic transaction* as discussed in claims 18 and 48.

Regarding claims 20 and 50, Long and Traversat, in combination, teach all of the claimed subject matter as discussed above with respect to claims 18 and 48, Long further discloses *the operation is performed idempotently with a network resource process* (Long, Col. 19, Lines 9-12).

Regarding claims 23 and 53, Long and Traversat, in combination, teach all of the claimed subject matter as discussed above with respect to claims 18 and 48, Traversat

Art Unit: 2162

further discloses the step of *storing the operation if the lock contention does not exist* (Traversat, FIG. 8).

Regarding claim 25, Long teaches all the claim subject matters as discussed above with respect to claim 24, but does not disclose the step of *performing lock contention handling for the first and second operation; storing the first and second operation of the atomic transaction in a log if a lock contention is not detected; and generating a lock contention notification if the lock contention is detected*. Traversat teaches a method for updating and managing a configuration database, and further discloses the step of *performing lock contention handling for the first and second operation; storing the first and second operation of the atomic transaction in a log if a lock contention is not detected; and generating a lock contention notification if the lock contention is detected* (Traversat, Col. 8, Line 3-Col. 9, Line 65). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to include the step of performing lock handling in order to execute the operation in a stable and correct state.

Regarding claim 31, Long teaches all the claim subject matters as discussed above with respect to claim 29, but does not disclose the step of *forming lock contention handling for the set of atomic transactions; and generating the lock contention notification if the lock contention is detected*. Traversat teaches a method for updating and managing a configuration database, and further discloses the step of *forming lock contention handling for the set of atomic transactions; and generating the lock contention notification if the lock contention is detected* (Traversat, Col. 8, Line 3-Col. 9, Line 65). Therefore, it would have been

Art Unit: 2162

obvious for one of ordinary skill in the art at the time the invention was made to include the step of forming lock handling in order to execute the operation in a stable and correct state.

Regarding claim 35, Long teaches all the claim subject matters as discussed above with respect to claim 33, but does not disclose *the database to detect lock contention; the configuration manager to generate a notification of the lock contention detected by the database; and the interface to display a message corresponding to the notification generated by the configuration manger*. Traversat teaches a method for updating and managing a configuration database, and further discloses *the database to detect lock contention; the configuration manager to generate a notification of the lock contention detected by the database; and the interface to display a message corresponding to the notification generated by the configuration manger* (Col. 8, Line 3-Col. 9, Line 58). Therefore, it would have been obvious for one of ordinary skill in the art at the time the invention was made to include the step of detecting lock, generating a notification and displaying a message in order to execute the operation in a stable and correct state.

Claims 6, 11, 17, 22, 27, 28, 42, 47 and 52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Long [USP 6,526,416 B1].

Regarding claims 6, 11, 17, 22, 42, 47 and 52, Long teaches all of the claimed subject matter as discussed above with respect to claims 1, 7, 12, 18, 37, 43 and 48, but does not explicitly teach *the operation is received from a first user concurrently with a second*

operation received from a second user. However, as disclosed by Long, the network is operated by a multitude of users (Col. 1, Lines 14-28). Therefore, two operations occurs at the same time is obvious. It would have been obvious for one of ordinary skill in the art at the time the invention was made to include the condition of concurrency into the technique in order to execute the operation in a stable and correct state.

Regarding claim 27, Long teaches all the claim subject matters as discussed above with respect to claim 24, but does not explicitly disclose *the first and second operation are received from a first user and a third operation of a second transaction is received from a second user.* However, as disclosed by Long, the network is operated by a multitude of users (Col. 1, Lines 14-28). Therefore, a transaction with a first and second operation from a user, and another transaction with a third operation from another user is obvious. It would have been obvious for one of ordinary skill in the art at the time the invention was made to include two transactions into the technique in order to execute the operation in a stable and correct state.

Regarding claim 28, Long teaches all the claim subject matters as discussed above with respect to claim 24, but does not explicitly disclose *the second operation is received from a fist user concurrently with a third operation received from a second user.* However, as disclosed by Long, the network is operated by a multitude of users (Col. 1, Lines 14-28). Therefore, two operations occurs at the same time is obvious. It would have been obvious for one of ordinary skill in the art at the time the invention was made to include

Art Unit: 2162

the condition of concurrency into the technique in order to execute the operation in a stable and correct state.

Conclusion

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to HUNG Q PHAM whose telephone number is 571-272-4040. The examiner can normally be reached on Monday-Friday.


If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, JOHN E BREENE can be reached on 571-272-4107. The fax phone

Art Unit: 2162

number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Examiner Hung Pham
April 15, 2005



SHAHID ALAM
PRIMARY EXAMINER